IN THE CLAIMS

Claims pending

- At time of the Action: Claims 1-74
- After this Response: Claims 1-40

Canceled or Withdrawn claims: 41-74

Amended claims: Claims 1 and 40

New claims: None

1. **(Currently Amended)** Interfaces, ~~embodied~~ stored on one or more computer-readable media, to be called on kernel transaction management objects, comprising:

application program interfaces (APIs) to implement operations on a kernel transaction object (TX), at least one TX representing a transaction and being accessible to at least one process participating in the transaction;

APIs to implement operations on a kernel resource management object (RMO), at least one RMO representing a relationship between a TX associated with a corresponding transaction manager and at least one resource that participates in the transaction; and

APIs to implement operations on a kernel enlistment (EN) object, at least one EN representing a relationship between a resource manager and the transaction.

2. **(Original)** Interfaces according to Claim 1, wherein each of the APIs to implement operations on TX, RMO, and EN utilize a handle to refer to an object.

3. **(Original)** Interfaces according to Claim 2, wherein each of the handles is an opaque reference to a unique object.

4. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for TX to transmit pre-prepare messages to resource managers associated with a transaction.

5. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for TX to transmit a prepare request to resource managers enlisted in a transaction.

6. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for a new TX to be created for a transaction.

7. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for an existing TX to be opened for a transaction.

8. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for TX to commit a transaction.

9. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for TX to abort a transaction.

10. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for TX to save a current state of the transaction.

11. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for TX to retrieve information about the TX for a requestor.

12. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for TX to set information.

13. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for TX to close.

14. **(Original)** Interfaces according to Claim 2, wherein at least one API is:

PreprepareEnlistment,

PrepareEnlistment,

OpenEnlistment

CreateTransaction,

OpenTransaction,

CommitTransaction,

RollbackTransaction,

SavepointTransaction,

GetTransactionInfo, and

SetTransactionInfo.

15. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for a new RMO to be created.

16. **(Original)** Interfaces according to Claim 15, wherein the new RMO is volatile.

17. **(Original)** Interfaces according to Claim 15, wherein the new RMO is durable.

18. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for an existing RMO to open for a transaction.

19. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for RMO to be destroyed.

20. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for RMO to transmit information regarding RMO to a requestor.

21. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for RMO to set information.

22. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for RMO to be enlisted on a transaction at least once.

23. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for a notification from a resource manager for RMO.

24. **(Original)** Interfaces according to Claim 2, wherein at least one API is:

CreateResourceManager,

OpenResourceManager,

DestroyResourceManager,

GetResourceManagerInfo,

SetResourceManagerInfo,

CreateEnlistment, and

GetNotificationResourceManager.

25. **(Original)** Interfaces according to Claim 2, wherein at least one API is to implement operations on TX by RMO.

26. **(Original)** Interfaces according to Claim 25, wherein the at least one API is to inform TX that pre-preparing is complete.

27. **(Original)** Interfaces according to Claim 25, wherein the at least one API is to inform TX that transaction preparation has been completed by a requested resource manager.

28. **(Original)** Interfaces according to Claim 25, wherein the at least one API is to inform TX that a resource manager has completed rolling back a transaction.

29. **(Original)** Interfaces according to Claim 25, wherein the at least one API is to inform TX that a resource manager has committed to a transaction.

30. **(Original)** Interfaces according to Claim 25, wherein the at least one API is:

PrePrepareComplete,

PrepareComplete,

RollbackComplete, and

CommitComplete.

31. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for a resource manager to be registered as a communications resource manager for a particular protocol.

32. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for a representation of a transaction to be serialized into a buffer.

33. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for information representing registered protocols to be serialized into a buffer.

34. **(Original)** Interfaces according to Claim 32, wherein at least one API calls for a transaction represented by the serialization be made available by a transaction management object.

35. **(Original)** Interfaces according to Claim 2, wherein at least one API calls for a transaction to be propagated to a destination using push-style propagation.

36. **(Original)** Interfaces according to Claim 35, wherein at least one API calls for the output of the API calls for the transaction to be propagated to a destination using push-style propagation to be retrieved.

37. **(Original)** Interfaces according to Claim 31, wherein at least one API is called when transaction propagation has been completed.

38. **(Original)** Interfaces according to Claim 31, wherein at least one API is called when requested transaction propagation has failed.

39. **(Original)** Interfaces according to Claim 2, wherein at least one API is:

RegisterProtocolAddressInformation,

MarshallTransaction,

GetProtocolAddressInformation,

PullTransaction,

PushTransaction,

PushTransactionBuffer,

PropagationComplete, and

PropagationFailed.

40. **(Currently Amended)** An apparatus for implementing a transaction, comprising:

a kernel transaction object (TX) to represent a transaction and being accessible to at least one process participating in the transaction;

a kernel resource manager object (RMO) to represent a relationship between a TX associated with a corresponding transaction manager and at least one resource that participates in the transaction; and

a kernel enlistment object (EN) to represent a relationship between a resource manager and the transaction,

wherein two-phase commit processing is executed by calling APIs on the TX, the RMO, and the EN.

41-74. **(Canceled)**